Fibre Channel Adapter for VMware ESX User's Guide

Document Revision History	
Revision A, January 8, 2009	
Changes	**Sections Affected**
New user's guide.	

Table of Contents

Glossary

Index

List of Figures

List of Tables

Preface

QLogic provides a portfolio of Fibre Channel (FC) adapters on the VMware® ESX/ESXi platform. The portfolio of QLogic FC adapters includes 2Gb, 4Gb, and 8Gb adapter offerings on VMware ESX/ESXi in PCI-X™, PCI Express®, and PCI Express ExpressModule™ form factors.

This guide provides detailed instructions on the installation, configuration, and troubleshooting of QLogic FC adapters on VMware ESX/ESXi. It also provides details on the use of QLogic FC adapter features to enhance the value of server virtualization using VMware ESX/ESXi. Such features include virtual adapter configuration using N_Port ID Virtualization (NPIV) and "boot from SAN" configuration.

Intended Audience

The *Fibre Channel Adapter for VMware ESX User's Guide* is intended for use by various consumers of QLogic FC adapters on VMware ESX/ESXi: users ranging from end users, such as data center managers and system administrators, to the test and development community. The information detailed in this guide is equally applicable to all types of QLogic FC adapter users on VMware ESX/ESXi.

Related Materials

For additional information, refer to the following documents:

- *SANsurfer FC HBA Manager User's Guide* (part number SN54651-00)
- SANsurfer FC HBA Manager release notes and ReadMe file
- *SANsurfer FC HBA CLI User's Guide* (part number SN0054614-00)
- SANsurfer FC HBA CLI release notes and ReadMe file
- VMware *IO Compatibility Guides*
- VMware technical note, *Configuring and Troubleshooting N-Port ID Virtualization*
- VMware *Fibre Channel SAN Configuration Guide*

What's in This Guide

This user's guide provides a brief introduction to the FC adapter offerings from QLogic at a broad level. The main focus of this guide is to explain the FC adapter software driver features, management tools available on VMware ESX/ESXi, and the supported features.

This preface covers the intended audience, related materials, document conventions used, license agreements, and technical support. The remainder of the guide is organized into the following chapters and appendices:

■ 1 Hardware and Software Components provides details about hardware components, software components, and the supported QLogic adapters.

■ 2 Installation details the supported features and covers installation of hardware, drivers, and SANsurfer® management tools.

■ 3 Configuration covers how to identify QLogic adapters, start *Fast!*UTIL, verify that the driver is loaded, configure driver parameters, deploy N_Port ID virtualization (NPIV), monitor NPIV from SANsurfer, and configure the boot from SAN feature.

■ A Troubleshooting details troubleshooting methods, including LEDs and logs.

At the end of this user's guide you will find a glossary of terms and an index to help you quickly find information.

Documentation Conventions

This guide uses the following typographic conventions:

NOTE:

Notes provide additional information.

■ Text in **bold font** indicates user interface elements such as a menu items, buttons, check boxes, or column headings. For example:

> Under **Hardware**, click **Storage Adapters**.

■ Text in a monospace font (`Courier New`) indicates file name, directory path, or command line text output. For example:

> `Enter an IP Address:`

■ Text in a bold monospace font (**`Courier New`**) indicates user input (what you type). For example:

> `# `**`chkconfig firewall off`**

- Text in ALL CAPITALS indicates the name of a key that you press on the keyboard. For example:

  On the QLogic BIOS banner screen, press CTRL+Q.

- Text in *italics* indicates emphasis, terms, or document titles. For example:

  Refer to the VMware *IO Compatibility Guide*.

- Italicized text within angle brackets indicates user-defined variables. For example:

  ```
  cat /proc/scsi/qla2300/<adapter_number>
  ```

- Text within square brackets indicates optional parameters. For example

  ```
  iscli [optional parameters] [Command Variable]
  ```

- Text in blue font indicates a hyperlink (jump) to a figure, table, chapter, or section in this guide, and links to Web sites are shown in underlined blue. For example:

  Table 4-1 lists the configurable driver parameters.

  See "Configuring Boot from SAN" on page 4-10.

  For more information, visit www.qlogic.com.

- Screen captures that depict only the essential portion of the user interface currently under discussion (rather than the entire dialog box or window) are shown with jagged edges, as shown in Figure i.
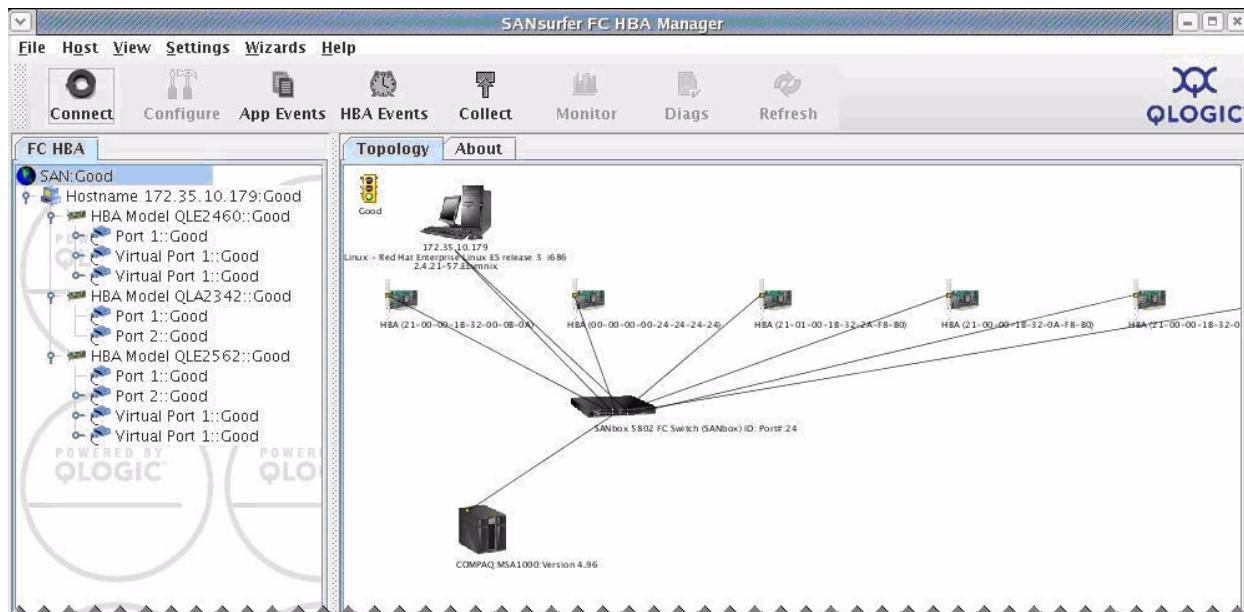

Figure i   Jagged-edged Screen Shots Depict Partial Interface

License Agreements

Refer to the *QLogic Software End User License Agreement* for a complete listing of all license agreements affecting this product.

Technical Support

Customers should contact their authorized maintenance provider for technical support of their QLogic switch products. QLogic-direct customers may contact QLogic Technical Support; others will be redirected to their authorized maintenance provider.

Visit the QLogic support Web site listed in Contact Information for the latest firmware and software updates.

Availability

QLogic Technical Support for products under warranty is available during local standard working hours excluding QLogic Observed Holidays.

Training

QLogic offers training for technical professionals for all iSCSI, InfiniBand, and Fibre Channel products. From the main QLogic Web page, www.qlogic.com, click the **Education and Resources** tab at the top, and then click the **Education & Training** tab at the left. The QLogic Global Training Portal offers online courses, certification exams, and scheduling of in-person training.

Technical certification courses include installation, maintenance, and troubleshooting QLogic products. Upon demonstrating knowledge using live equipment, QLogic awards a certificate identifying the student as a Certified Professional. The training professionals at QLogic may be reached by e-mail at training@qlogic.com.

Contact Information

Please feel free to contact your QLogic approved reseller or QLogic Technical Support at any phase of integration for assistance. QLogic Technical Support can be reached by the following methods:

Web	http://support.qlogic.com
E-mail	support@qlogic.com

The QLogic knowledge database contains troubleshooting information for the QLogic adapters. Access the database from the QLogic Support Web page, http://support.qlogic.com. Use the Support Center search engine to look for specific troubleshooting information.

1 Hardware and Software Components

This chapter provides details about QLogic FC adapters for VMware, including:

-
-
-

Hardware Components

A typical QLogic FC adapter includes the following major components:

- RISC processor
- Static random access memory (SRAM)
- Non-volatile memory (Flash)
- Non-volatile random access memory (NVRAM) (On ISP25*xx*, NVRAM is built into the Flash)
- Transceivers
- LEDs

Figure 1-1 shows the hardware components of a QLogic FC adapter and Table 1-1 provides additional details of selected components.

5 **NVRAM:** Non-Volatile Random Access Memory (aka EEPROM). Where some of the adapter parameter settings are stored. Because NVRAM is a non-volatile type of RAM, data is retained even if the power is shut off.

1 **Processor:** QLogic's ISP contains a RISC processor.

2 **SRAM:** Static Random Access memory. Where the firmware is loaded. In addition to the firmware, SRAM also retains (in queue) the various I/O commands the FC processor must process.

Transceivers: Short for transmitter-receiver, a device (copper or optical) that transmits and receives signals.

4 **LEDs:** Light Emitting Diodes. Display various active/inactive states of an adapter. QLogic's Beaconing feature uses LEDs to identify the physical location of the adapter.

3 **Flash:** Non-volatile form of memory. Where the boot code is stored.

Figure 1-1 FC Adapter Components

Table 1-1. FC Adapter Component Details

Callout Number	Description
1	The reduced instruction set computer (RISC) processor, also known as the ISP controller, is the heart of the adapter. The RISC processor is a single integrated ASIC (application specific integrated circuit) that contains the processing power required to run the FC protocol. The RISC processor also interfaces the host bus (PCI-X or PCI Express) and controls the I/O operations.
2	The static random access memory (SRAM) provides the runtime memory required for the adapter operations. The SRAM is where the firmware is downloaded when the adapter is powered up. The RISC processor (see *1*) accesses the SRAM to process the firmware instructions. In addition, the SRAM stores the runtime context of various I/O operations in progress.
3	The non-volatile memory, also referred to as Flash memory, contains the boot code run by the RISC processor (see *1*) when the adapter is powered up. On QLogic FC adapter 2400 series and later, the Flash memory contains multiple boot images—the FCode, x86 BIOS, and EFI—to support multi-boot capability on different platforms.
4	LEDs indicate the real-time status of the adapter, FC link status, and link activity. LED patterns are described in detail in "Troubleshooting with LEDs" on page A-1.

Table 1-1. FC Adapter Component Details (Continued)

Callout Number	Description
5	Non-volatile random access memory (NVRAM) is used to store the per-port adapter configuration parameters. These parameters are typically loaded at manufacturing and should not be modified except by those who have an expert level understanding of the functioning of QLogic FC adapters. During adapter initialization, the parameters are accessed and used to configure the adapter. Examples of adapter configuration parameters include FC connection type (topology), FC data rate, FC tape support, and so on.

Software Components

This section describes the following QLogic FC adapter software components:

- Boot Code
- OS Driver
- Firmware
- Management Tools and Utilities

Boot Code

Boot code resides on the Flash memory on the adapter. The primary function of the boot code is to communicate with an external boot device before the operating system takes the control over when booting from a storage device attached to the QLogic adapter. This type of configuration is referred to as *boot from SAN*.

OS Driver

For all QLogic FC adapters currently supported on VMware ESX/ESXi, the driver is supported natively with ESX; that is, no separate driver installation is required for VMware ESX/ESXi to recognize the QLogic FC adapters. QLogic FC adapters on VMware ESX/ESXi require no additional driver installation steps to run the QLogic FC adapter.

Firmware

The firmware module performs most of the FC protocol-related functions, thus freeing the system CPU and memory resources from protocol processing. This efficiency keeps the QLogic FC driver *thin*, which saves runtime system resources such as CPU and memory.

The QLogic FC adapter firmware is bundled with the driver. This unified approach ensures that the driver and firmware are always in sync on a customer system and simplifies the manageability of QLogic FC drivers.

Management Tools and Utilities

QLogic offers a comprehensive suite of management tools and utilities, including the following:

■ **SANsurfer FC HBA Manager** provides a graphical user interface for simplified management.

■ **SANsurfer FC HBA CLI** (command line interface) enables you to automate FC adapter management by means of scripting and interactive mode.

QLogic FC Adapters Supported on VMware ESX/ESXi

For a complete list of QLogic FC adapters supported on VMware ESX/ESXi, refer to the VMware *IO Compatibility Guides* located as follows:

■ VMware ESX 3.5:

http://www.vmware.com/pdf/vi35_io_guide.pdf

■ ESX Server 3.0.x:

http://www.vmware.com/pdf/vi3_io_guide.pdf

For the latest list of supported platforms and I/O devices, refer to the respective VMware compatibility guides for your VMware ESX version at the following URL:

http://www.vmware.com

2 Installation

This chapter provides the following installation information:

-
-
-
-

Supported Features

When properly installed, QLogic FC adapters for VMware ESX support the following features:

- **Topologies**—FC-AL, FC-AL2, point-to-point, and switched fabric.
- **Protocols**—FCP-3-SCSI, FC-tape (FCP-2).
- **Link Rate**—2Gb, 4Gb, and 8Gb FC adapters with auto-rate negotiation.
- **SNIA Common adapter API**—v1.x/v2.0 compliant.
- **QLogic CMPI providers**—for ESXi 3.5 U2 and later.
- **N_Port ID virtualization**—inbox driver supports 63 virtual adapters per port.
- **Remote SAN and adapter management**—using the SANsurfer FC HBA Manager tool (see "Installing SANsurfer FC HBA Manager" on page 2-3).
- **Automated local host management**—using the SANsurfer FC HBA CLI tool (see "Installing SANsurfer FC HBA CLI" on page 2-6).

Installing the Hardware

Adapter hardware installation requires that you open the computer and locate the appropriate bus slot for housing the adapter, as detailed in the following steps. Figure 2-1 shows the hardware placement.
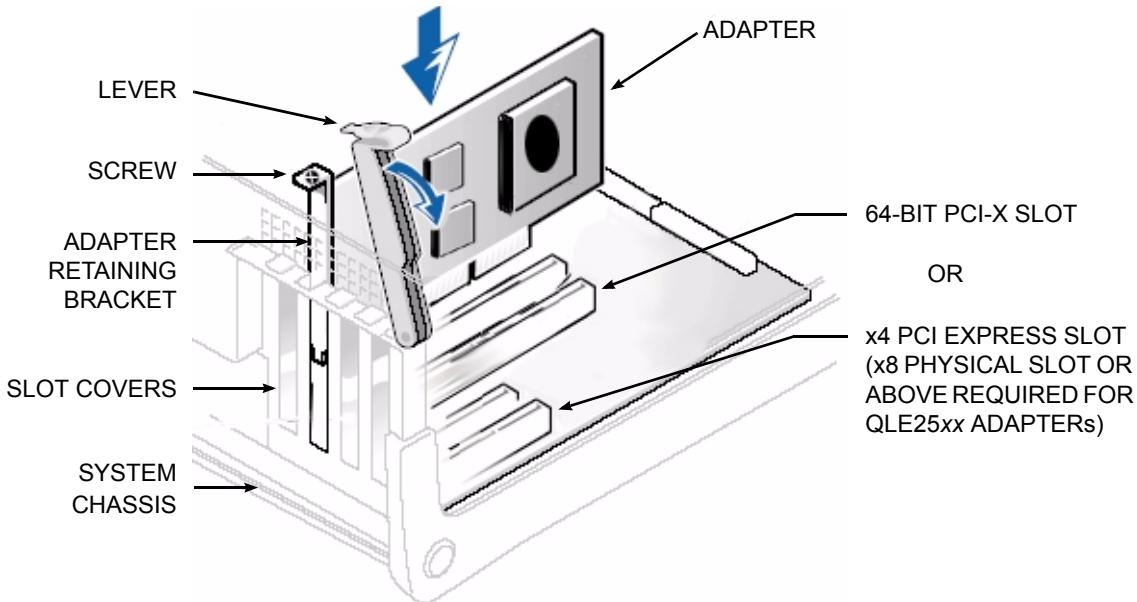

Figure 2-1 Installing the Adapter

To install the adapter:

1.   Power off the computer, and then disconnect the power cable.

2.   Remove the computer cover and find an appropriate empty bus slot.

3.   Pull out the slot cover (if any) by removing the screw or releasing the lever.

4.   Grasp the adapter by the top edge and seat it firmly into the appropriate slot.

5.   Refasten the adapter's retaining bracket using the existing screw or lever.

6.   Close the computer cover.

7.   Plug the FC cable into the adapter.

8.   Plug in the power cable and turn on the computer.

9.   Verify the LEDs on the adapter to ensure proper operation. The LED patterns are described in "Troubleshooting with LEDs" on page A-1.

Installing the Driver

The QLogic FC driver ships "inbox" with VMware ESX/ESXi. VMware's patch process issues all driver updates. For details, refer to the VMware support Web site.

Installing SANsurfer Management Tools

The QLogic FC management tools include the SANsurfer FC HBA Manager (GUI) and the SANsurfer FC HBA CLI (command line interface).

■ **SANsurfer FC HBA Manager** provides:

❑ A graphical user interface to easily install, configure, and deploy QLogic FC adapters.

❑ Diagnostic and troubleshooting capabilities.

❑ Statistical information required to optimize SAN performance.

❑ Remote management capabilities.

■ **SANsurfer FC HBA CLI** provides:

❑ Management of QLogic FC adapters only on the local machine on which the tool is installed.

❑ A non-interactive mode that can be used for scripting common and frequent operations.

❑ Menu-driven mode for ease of use.

Installing SANsurfer FC HBA Manager

To run SANsurfer FC HBA Manager on a VMware ESX server requires a two-step installation process, whereby:

1. You must install the SANsurfer back end, commonly called the *qlremote agent*, on the ESX server.

2. You must download the SANsurfer FC HBA Manager GUI from the QLogic Web site or CD-ROM and install it on a Windows® or Linux® system. SANsurfer FC HBA Manager allows you to remotely manage QLogic adapters.

After SANsurfer FC HBA Manager is installed, you must connect it to the qlremote agent using the IP address or host name of the ESX host on which the qlremote agent is installed. When SANsurfer FC HBA Manager is connected to the qlremote agent, you can use it to view and manage the QLogic adapters.

NOTE:

> The qlremote agent requires a service called *portmap* running on the ESX host. To turn on the portmap service, execute the following commands on the ESX Server Console:

```
# chkconfig firewall off
# chkconfig iptables off
# chkconfig -level 3 portmap on
# service firewall restart
# service iptables restart
# service portmap restart
```

SANsurfer FC HBA Manager requires that you select the download link to the package for your operating system, either from an installation CD or the QLogic Web site.

To install SANsurfer FC HBA Manager on VMware:

1. Go to the QLogic **Driver Downloads / Documentation** page at the following URL:

    http://driverdownloads.qlogic.com/QLogicDriverDownloads_UI/default.aspx

2. Click the **by Operating System** button, and then select **Fibre Channel HBAs**, **VMware**, and **VMware ESX/ESXi**, as shown in Figure 2-2.
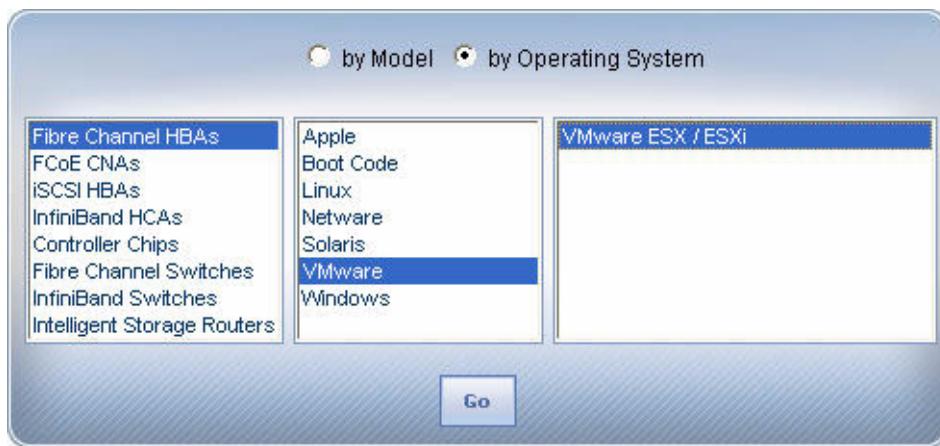

Figure 2-2 Downloading SANsurfer FC HBA Manager—VMware

3. Click **Go**.

4. Under **Management Tools**, click **SANsurfer FC HBA Manager (x86/x64)**.

5. On the End User Software License Agreement page, click **Agree**.

6.   Save the zipped file and then use it to install SANsurfer FC HBA Manager by following the installation wizard.

7.   Install the qlremote agent on your ESX host using silent mode as follows:

```
# ./standalone_sansurfer5.0.1b33_linux_install.bin -i silent
```

To install SANsurfer FC HBA Manager on Linux or Windows:

1.   Go to the QLogic **Driver Downloads / Documentation** page at the following URL:

http://driverdownloads.qlogic.com/QLogicDriverDownloads_UI/default.aspx

2.   Click the **by Operating System** button, and then select **Fibre Channel HBAs**, **Linux** (or **Windows**) OS, and the specific OS version. Figure 2-3 shows an example of selecting Linux Red Hat® 64-bit.
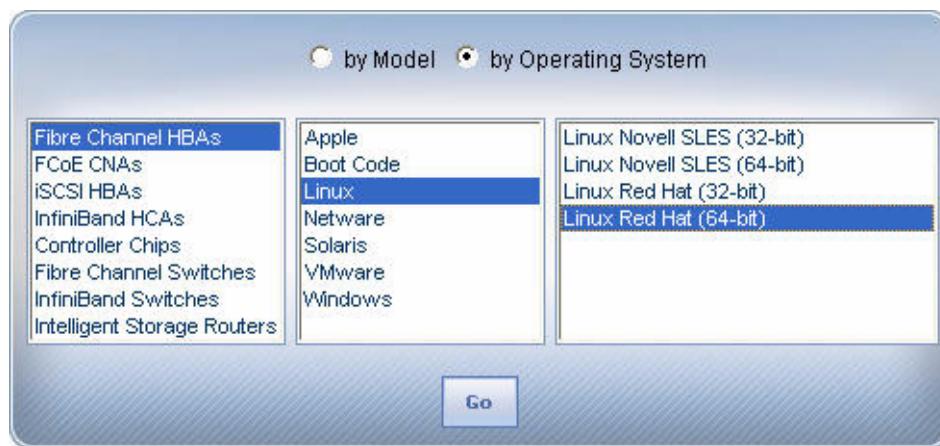
Figure 2-3 Downloading SANsurfer FC HBA Manager—Linux

3.   Click **Go**.

4.   Under **Management Tools**, click **SANsurfer FC HBA Manager (x86/x64)**.

5.   On the End User Software License Agreement page, click **Agree**.

6.   Save the zipped file and then use it to install SANsurfer FC HBA Manager by following the installation wizard.

After you complete the preceding installation steps, you must connect SANsurfer FC HBA Manager from the Linux or Windows host to the qlremote agent running on the ESX host.

The default password for SANsurfer FC HBA Manager is *config*. To ensure that security is not compromised, change this password as soon as possible after installing SANsurfer.

Installing SANsurfer FC HBA CLI

Follow these steps to download and install the SANsurfer FC HBA CLI utility.

To install SANsurfer FC HBA CLI:

1. From the QLogic Web site or the SANsurfer Management Suite CD-ROM, download the `scli-x.xx.xx-xx.i386.rpm.gz` file to your local machine. (In the file name, `x.xx.xx` indicates the current version of SANsurfer FC HBA CLI.

2. To uncompress the file, execute the following command:

    `# gunzip scli-x.xx.xx-xx.i386.rpm.gz`

3. To install the package, enter the following command:

    `# rpm -iv scli-x.xx.xx-xx.i386.rpm`

    The new package is installed in the following directory:

    `/opt/QLogic_Corporation/SANsurferCLI`

4. To query for the installed SANsurfer FC HBA CLI package name, enter the following command from the command line:

    `# rpm -q scli`

3 Configuration

QLogic FC adapters on VMware ESX configuration procedures include the following:

- "Identifying QLogic FC Adapters on VMware ESX" on page 3-1
- "Using the Fast!UTIL BIOS Utility" on page 3-4
- "Verifying That the Driver is Loaded" on page 3-5
- "Configuring the Driver Parameters" on page 3-6
- "Deploying NPIV for VMware ESX/ESXi" on page 3-9
- "Monitoring NPIV Virtual Ports from SANsurfer" on page 3-9
- "Configuring Boot from SAN" on page 3-11

Identifying QLogic FC Adapters on VMware ESX

On VMware ESX, you can identify QLogic FC adapters in two ways:

- Using VI Client
- Using Console OS

Using VI Client

You can use the VM Infrastructure Client (VI Client) to identify QLogic FC adapters on a VMware ESX system.

To identify QLogic FC adapters using VI Client:

1. Start VM Infrastructure Client and select the host you want.

2. Click the **Configuration** tab to bring the Configuration page to the front.

3. Under **Hardware** on the left, click **Storage Adapters**.

   A list of all storage adapters configured on the system appears, including QLogic FC adapters, as shown in Figure 3-1.
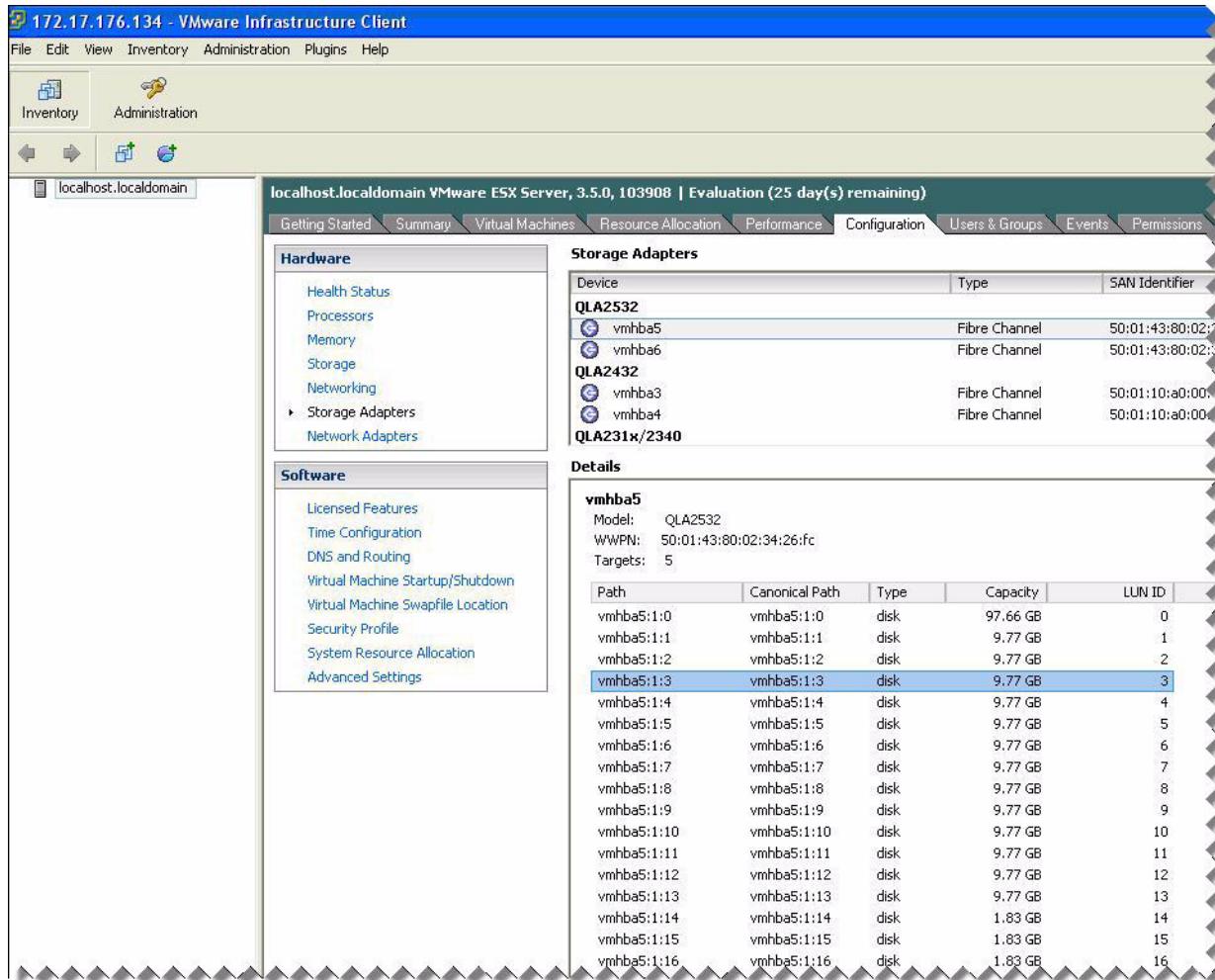
Figure 3-1 VI Client Configuration Page

Using Console OS

You can use the console OS to identify QLogic FC adapters on a VMware ESX system.

To identify QLogic FC adapters using console OS:

1.   To list the subdirectories, run the following command:

    `ls /proc/scsi:`

    If the `qla2300` directory is present, QLogic adapters are installed on the VMware ESX system.

2. To list the QLogic storage adapters present on the system, run the following command:

    `ls /proc/scsi/qla2300`

    An example output may be:

    ```
    # ls /proc/scsi/qla2300/
    1  2  3  4  5  6   HbaApiNode
    ```

3. To view additional details, run the following command:

    `cat /proc/scsi/qla2300/<``adapter_number`**`>`**

    Where **`<`**`adapter_number`**`>`** is one of the adapter numbers shown in Step 2.

    This command lists the firmware revision, driver revision, and the boot code version, as shown in Figure 3-2.
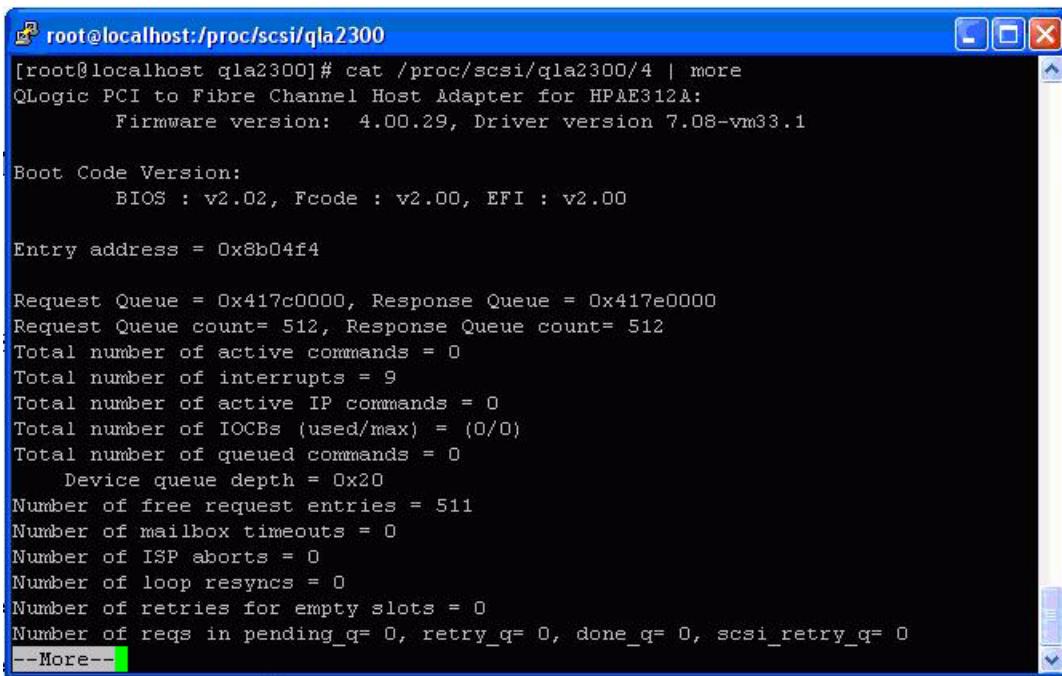
```
root@localhost:/proc/scsi/qla2300
[root@localhost qla2300]# cat /proc/scsi/qla2300/4 | more
QLogic PCI to Fibre Channel Host Adapter for HPAE312A:
        Firmware version:  4.00.29, Driver version 7.08-vm33.1

Boot Code Version:
        BIOS : v2.02, Fcode : v2.00, EFI : v2.00

Entry address = 0x8b04f4

Request Queue = 0x417c0000, Response Queue = 0x417e0000
Request Queue count= 512, Response Queue count= 512
Total number of active commands = 0
Total number of interrupts = 9
Total number of active IP commands = 0
Total number of IOCBs (used/max) = (0/0)
Total number of queued commands = 0
    Device queue depth = 0x20
Number of free request entries = 511
Number of mailbox timeouts = 0
Number of ISP aborts = 0
Number of loop resyncs = 0
Number of retries for empty slots = 0
Number of reqs in pending_q= 0, retry_q= 0, done_q= 0, scsi_retry_q= 0
--More--
```

Figure 3-2 Finding Adapters on VMware ESX

Using the *Fast!*UTIL BIOS Utility

QLogic FC adapters provide a BIOS utility, *Fast!*UTIL, which you can invoke at the BIOS. QLogic FC adapters support the *Fast!*UTIL BIOS utility on VMware ESX systems. *Fast!*UTIL does not require any installation.

Use *Fast!*UTIL to perform the following operations:

- Record BIOS, WWPN, and serial number information
- Restore factory defaults
- Scan for fibre devices
- Configure boot from SAN device
- Perform low-level disk commands
- Loopback test with loopback plug

To invoke *Fast!*UTIL, press the CTRL+Q keys at the QLogic BIOS banner screen, as described in steps 1 and 2 of "Configuring Boot from SAN" on page 3-11.

The *Fast!*UTIL utility provides the following options:

- **Configuration Settings** provide options for Host Adapter Configuration, Boot Device Configuration, and an option to restore Default Configuration.

- **Scan Fibre Channel Devices** option scans the FC loop and lists all the connected devices by loop ID. This option also displays device information such as vendor ID, product ID, and revision.

- **Fibre Disk Utility** scans the FC loop and provides options to verify disk media, verify disk data, and format the FC disk devices.

- **Loopback Data Test** runs a loopback test to ensure that the loop is up and to verify the health of the FC loop.

- **Select Host Adapter** setting enables you to select a specific adapter, if you have multiple QLx23*xx* adapters in your system, and then configure or view the adapter's settings.

The *Fast!*UTIL BIOS utility is described in detail in the *BIOS for 2Gb FC HBAs Readme File*, located at the following URL:

http://filedownloads.qlogic.com/webservices_Dev/JK16E11E4F75D65007A771F6F2746AE2B42D/file0001-readme_bios_2gb_154.pdf

Verifying That the Driver is Loaded

Follow this procedure to ensure that the QLogic FC adapter driver is loaded on the ESX system

**To verify that the QLogic FC adapter driver is loaded:**

1. Log in to the service console as *root* user.

2. Execute the following command:

    `# esxcfg-module -l | grep qla2300`

    Depending on the ESX Server version, the listed module will be one of the following:

    ■ `qla2300_707` on ESX 3.0.*x*

    ■ `qla2300_707_vmw` on ESX 3.5 and later

    Figure 3-3 shows an example of the output. For proper operation, ensure that the QLogic driver shows as `Enabled Loaded`.
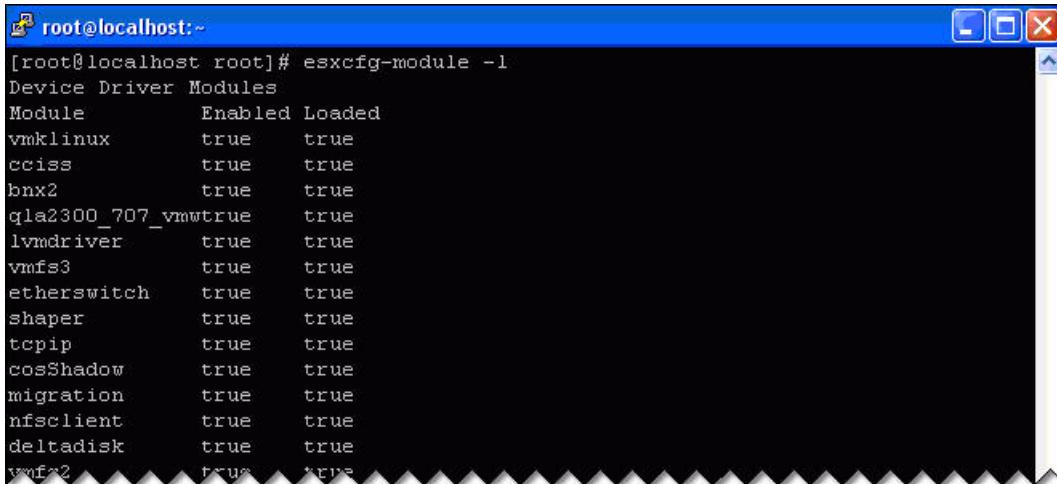
Figure 3-3 Verifying That the Driver is Loaded

Configuring the Driver Parameters

Use the `esxcfg-module` command to configure the QLogic FC adapter driver parameters.

To configure driver parameters:

1. To list the driver parameters, execute the following command:

    # **esxcfg-module -g qla2300_707_vmw**

2. To set the driver parameters, execute the following command:

    # **esxcfg-module -s <*options*> qla2300_707_vmw**

    where <*options*> is one of the configurable parameters listed in Table 3-1 on page 3-7.

    For example, the following command sets the maximum device queue depth reported by the QLogic to a value of 32:

    # **esxcfg-module -s ql2xmaxqdepth=32 qla2300_707_vmw**

3. To update the ESX bootstrap image, execute following command:

    # **esxcfg-boot -b**

4. For the preceding changes to take effect, reboot the server as follows:

    # **reboot**

NOTE:

All configuration changes made using the **esxcfg-module -s** command are *persistent*, meaning that the configuration changes will be saved across system reboots.

Table 3-1 lists all the QLogic driver configurable parameters and provides a brief description of each.

Table 3-1. Configurable Driver Parameters

Driver Parameter	Description
`Bind`	Target persistent binding method. `0` = bind by port name (default); `1` = bind by port ID.
`extended_error_logging`	Option to enable extended error logging. `0` = no logging (default); `1` = log errors.
`max_srbs`	Maximum number of simultaneous commands allowed for an adapter.
`qfull_retry_count`	Number of retries to perform on `Queue Full` status from device. Default is `16`.
`qfull_retry_delay`	Number of seconds to delay on `Queue Full` status from device. Default is `2`.
`ql2xallocfwdump`	Option to enable allocation of memory for a firmware dump during adapter initialization. Memory allocation requirements vary by ISP type. `0` = memory not allocated (default).
`ql2xcmdtimermin`	Minimum command timeout value. Default is `30`.
`ql2xdevflag`	Set to `1` to display abbreviated persistent binding statements.
`ql2xfdmienable`	Enables FDMI registrations. `0` = no FDMI (default); `1` = perform FDMI.
`ql2xfwloadbin`	Option to specify to load ISP24*xx* firmware from Flash or embedded in driver. `1` = load firmware from Flash. `0` = load firmware embedded with driver (default).
`ql2xintrdelaytimer`	Waiting time for firmware before it generates an interrupt to the host to notify completion of request.
`ql2xioctltimeout`	Ioctl timeout value in seconds for pass-through commands. Default is `66`.
`ql2xlogintimeout`	Login timeout value in seconds. Default is `20`.
`ql2xmaxqdepth`	Maximum queue depth to report for target devices. Default is `32`.
`ql2xmaxsectors`	Maximum sectors per I/O request. Default is `512`.
`ql2xmaxsgs`	Maximum scatter or gather entries per I/O request. Default is `32`.
`ql2xoperationmode`	Option to enable operation mode for ISP23*xx* or ISP24*xx*. For ISP23*xx*, set to `0x5` to enable zero input output (ZIO) mode. For ISP24*xx*, set to `0x5/0x6` to enable ZIO mode.
`ql2xplogiabsentdevice`	Option to enable PLOGI (port login) to devices that are not present after a fabric scan. This workaround is required for some FC switch defects. `0` = no PLOGI (default); `1` = perform PLOGI.

Table 3-1. Configurable Driver Parameters (Continued)

Driver Parameter	Description
ql2xprocessnotready	Option to disable handling of NOT-READY status in the driver. **1** = handled by the driver (default); **0** = disable the handling inside the driver.
ql2xretrycount	Maximum number of mid-layer retries allowed for a command. Default value in non-failover mode is **20**, and in failover mode is **30**.
ql2xsuspendcount	Number of 6-second suspend iterations to perform while a target returns a <NOT READY> status. Default is **10** iterations.
ql2xusedefmaxrdreq	Option to adjust PCIe™ Maximum Read Request Size. **1** = use system default; **0** = default.
qla_cmd_timeout	Timeout value in seconds for scsi command. Default is **20**.
qllink_down_timeout [a]	Delay in seconds after a loop down event before the loop is flagged as down by the driver and the I/O requests are aborted. Default is **30** seconds.
qlogin_retry_count	Option to modify the login retry count.
qlport_down_retry [a]	Maximum number of command retries to a port that returns PORT-DOWN status. Default is **15**. (This parameter is multiplied by a factor of 2 to determine the number of seconds driver holds the I/O request.)
recoveryTime	Recovery time in seconds before a target device is sent I/O after a failback is performed.
retry_gidpt	Number of times GID_PT to be retried to get the node name and port ID of the device list.

Table Footnotes

[a] This parameter determines how long the driver will internally queue I/O before returning the **DID_NO_CONNECT** status to the SCSI mid layer.

Deploying NPIV for VMware ESX/ESXi

QLogic FC adapters support N_Port ID virtualization (NPIV) for FC SANs. NPIV enables each virtual machine to have its own FC world wide port name (WWPN) by creating multiple virtual adapters on a single physical adapter.

VMware ESX 3.5/3i enables NPIV support on QLogic FC adapters. Because the QLogic FC driver shipped inbox on VMware ESX/ESXi supports NPIV, no additional software or hardware installation is required. Note that VMware ESX/ESXi 3.5 supports NPIV only on Raw Disk Map devices.

To configure NPIV, the FC switch port must be configured to enable NPIV. The specific procedure to enable NPIV on the switch port depends on the switch itself. For details, refer to the FC switch documentation.

The VMware technical note, *Configuring and Troubleshooting N-Port ID Virtualization*, describes NPIV deployment for VMware ESX in greater detail. This document is available on the VMware Web site at the following URL:

http://www.vmware.com/pdf/vi3_35_25_npiv_config.pdf

Monitoring NPIV Virtual Ports from SANsurfer

QLogic's SANsurfer utility also supports the NPIV featured drivers. SANsurfer FC HBA Manager (GUI) and SANsurfer FC HBA CLI (command line interface) both display the virtual ports configured on the system. These utilities also provide additional information on the virtual ports, including diagnostics and monitoring information.

SANsurfer generates events and notifies the user of any change (such as virtual port creation and deletion) at the backend to the virtual ports.

Figure 3-4 shows a typical virtual environment in SANsurfer FC HBA Manager with physical and virtual ports.
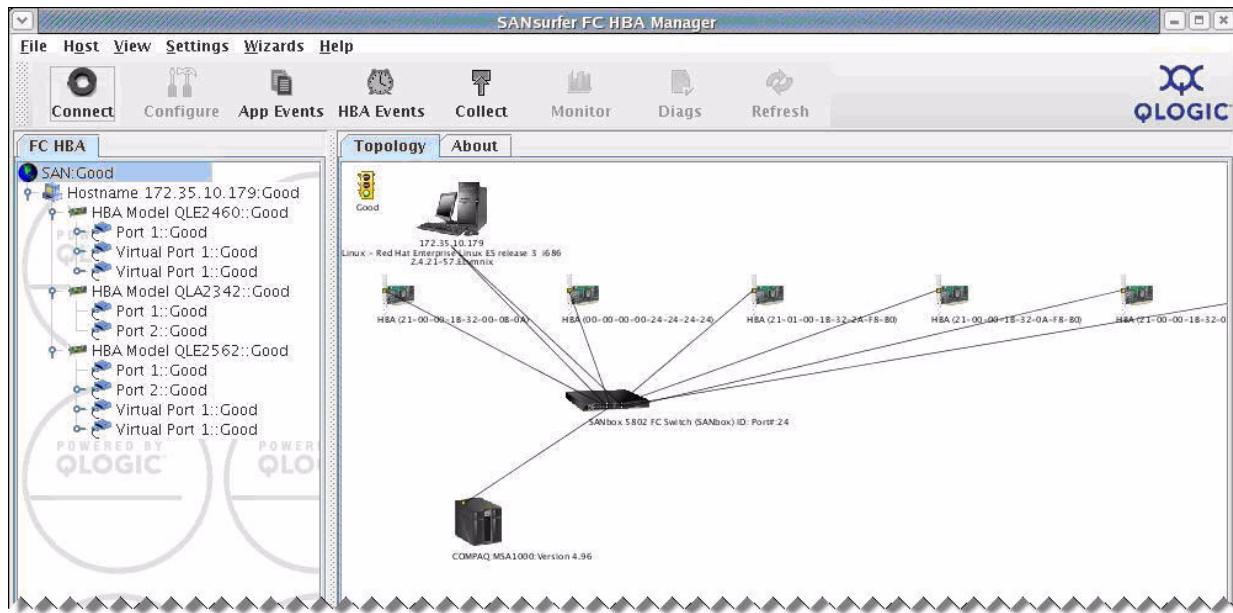
Figure 3-4 SANsurfer FC HBA Manager Showing NPIV

When you select an individual virtual port on the Topology page, SANsurfer FC HBA Manager displays additional detail about the selected virtual port on four tabbed pages. The Information page (see Figure 3-5) displays port attributes such as **Port ID**, **Firmware Version**, and **Actual Data Rate**.
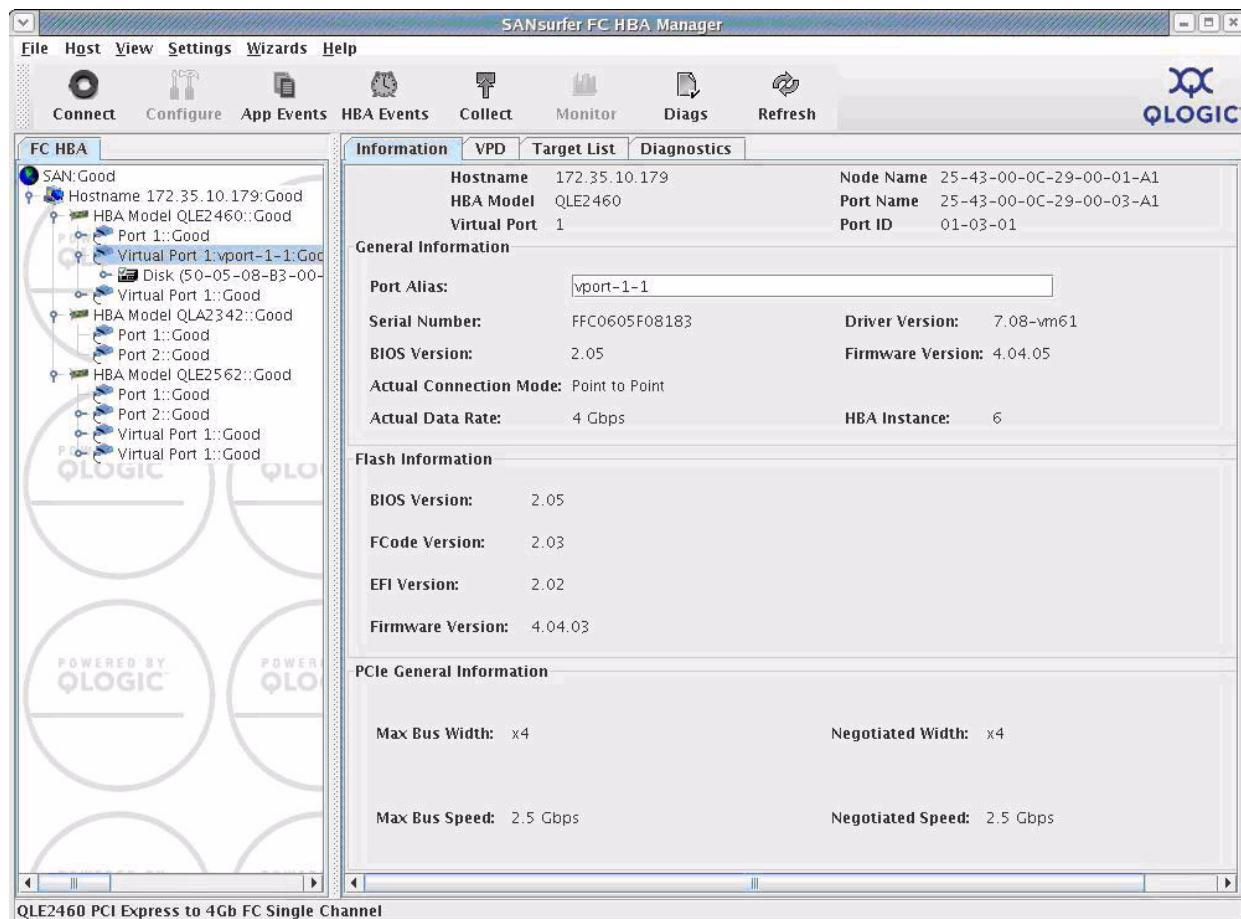
Figure 3-5 Virtual Port Information Page in SANsurfer

Configuring Boot from SAN

Boot from SAN refers to installing and booting a server's operating system from a storage device attached to the SAN instead of from a locally attached storage device. The OS is installed on one or more LUNs in the SAN array and the FC adapters are configured to boot from the specified LUN. Boot from SAN providers several benefits including:

- More affordable servers
- Ease of management
- Improved disaster tolerance
- Improved boot LUN availability

VMware ESX 3.*x* provides the boot from SAN capability on QLogic FC adapters. For storage array and FC switch configuration, refer to the appropriate user manual for the device. For VMware ESX-specific configuration requirements, refer to the VMware *Fibre Channel SAN Configuration Guide* located at the following URL:

https://www.vmware.com/pdf/vi3_35/esx_3/r35/vi3_35_25_san_cfg.pdf

To configure the QLogic FC adapter for boot from SAN:

1.   After installing the adapter, boot the server into BIOS.

2.   At the QLogic BIOS banner screen (Figure 3-6), press the CTRL+Q keys.

Figure 3-6 Starting Fast!UTIL from the BIOS Screen

The *Fast!*UTIL FC adapter configuration utility starts.

3.   In *Fast!*UTIL, choose the appropriate FC adapter port to be configured.

4.   On the *Fast!*UTIL Options page, select **Configuration Settings**, and then press ENTER.

5.   On the Configuration Settings page, select **Host Adapter Settings**, and then press ENTER.

6.   To set the BIOS to search for SCSI devices:

a.   On the Host Adapter Settings page, select **Host Adapter BIOS**.

b.   To toggle the value to **Enabled**, press the ENTER key.

c.   To exit, press the ESC key.

7.   After the **Host Adapter BIOS** is enabled, select the boot device as shown in the following table:

Enable Selectable Boot	WWPN/LUN Boot List	Device Boot
No	x	BIOS configures the first disk drive it finds as boot device.
Yes	None specified	BIOS configures the first disk drive it finds that is also a LUN 0 as boot device.
Yes	Specified	BIOS scans through the specified boot WWPN or LUN list until it finds a disk drive, and then configures it as the boot device.

    a.   Choose **Selectable Boot Settings**, and then press ENTER.

    b.   On the Selectable Boot Settings page, choose **Selectable Boot**.

    c.   To toggle the value to `Enabled`, press ENTER.

8.   After the Selectable Boot is enabled, select the boot device. QLogic FC adapters allow the specification of up to four WWPN/LUNs as boot devices. To select up to three boot devices, follow these steps:

    a.   To move to the **Primary** location of the **Selectable Boot List** menu, press the DOWN ARROW key.

    b.   To see a list of accessible devices in the **Select Fibre Channel Device** menu, press ENTER.

    c.   Press the DOWN ARROW key to scroll down to the device you want to put into your **Selectable Boot** menu list.

    d.   To select the requested device and load it into the **Selectable Boot** menu list, press ENTER.

    e.   (Optional) Repeat this step to specify up to three alternate boot devices.

Now the ESX system is ready to boot from SAN from the specified boot devices.

NOTE:

QLogic recommends that when both the adapter BIOS and the **Selectable Boot** option are enabled, you always select a device and put it in the **Primary** boot device location of the **Selectable Boot** menu list.

Notes

A Troubleshooting

This appendix provides the following methods of determining the status of the adapters and finding solutions to common problems:

Troubleshooting with LEDs

QLogic FC adapters have LEDs that can help you diagnose problems. Typically, administrators observe the LED patterns to perform first-level troubleshooting that can indicate some obvious firmware or hardware issues.

Specific QLogic FC adapter families have unique LED patterns. The following tables describe the LED patterns for the various FC adapter series.

Table A-1. QLA2310/QLA200 Adapter LED Scheme

Green LED	Yellow LED	Activity
❍	❍	Power off
●	●	Power
●	❍	Online
❍	●	Link
❍	Flashing	Loss of synchronization
Green and yellow LEDs flashing alternately		Firmware error

Table A-2. QLA234x/QLA210/QLE220 Adapter LED Scheme

Green LED	Amber LED	Activity
❍	❍	Power off
●	●	Power
●	❍	Online
❍	●	Link
❍	Flashing	Loss of synchronization
Alternate flashing	Alternate flashing	Firmware error
Flashing	❍	Beacon

Table A-3. QLA24xx/QLE24xx/QEM2462 Adapter LED Scheme

Yellow LED	Green LED	Amber LED	Activity
❍	❍	❍	Power off
●	●	●	Power on (before firmware init)
Flashing	Flashing	Flashing	Power on (after firmware init)
Yellow, green, and amber LEDs flashing alternately			Firmware error
❍	❍	●/Flashing	Online, 1Gbps link / I/O activity
❍	●/Flashing	❍	Online, 2Gbps link / I/O activity
●/Flashing	❍	❍	Online, 4Gbps link / I/O activity
Flashing	❍	Flashing	Beacon

Table A-4. QEM2462 Power and Attention Switch LED Scheme

Green Power	Activity
○	Power off Insertion or removal of the adapter is permitted
●	Power Insertion or removal of the adapter is *not* permitted
Blinking	Hot-plug operation is in progress and insertion or removal of the adapter is *not* permitted

Yellow Attention	Activity
○	Normal operation
●	Operational problem at this slot
Blinking	Slot is being identified at the user's request

Table A-5. QLE25xx Adapter LED Scheme

Yellow LED (8Gbps)	Green LED (4Gbps)	Amber LED (2Gbps)	Activity
○	○	○	Power off
●	●	●	Power on (before firmware init)
Flashing	Flashing	Flashing	Power on (after firmware init)
Yellow, green, and amber LEDs flashing alternately			Firmware error
○	○	●/Flashing	Online, 2Gbps link / I/O activity
○	●/Flashing	○	Online, 4Gbps link / I/O activity
●/Flashing	○	○	Online, 8Gbps link / I/O activity
Flashing	○	Flashing	Beacon

Dynamic Extended Logging

The QLogic driver provides a mechanism to dynamically enable extended error logging about driver operation. These logs provide debugging and error information to help you understand driver behavior regarding issues related to the adapter. These logs are saved in a system log file named `vmkernel`, in the `/var/log/` directory.

To enable extended error logging, execute the following command:

```
# echo "scsi-qlaenable-log" > /proc/scsi/qla2300/<adapter_number>
```

To disable extended error logging, execute the following command:

```
# echo "scsi-qladisable-log" > /proc/scsi/qla2300/<adapter_number>
```

When the logging is enabled, the driver version in the `/proc/scsi/qla2300/<adapter number>` is appended with a `-debug` flag as shown below:

```
# cat /proc/scsi/qla2300/5
QLogic PCI to Fibre Channel Host Adapter for QLE2462:
        Firmware version: 4.04.05, Driver version 7.08-vm62-debug
Boot Code Version:
        BIOS : v2.02, Fcode : v2.00, EFI : v2.00
Entry address = 0x8b0000
```

NOTE:

Although the driver logs provide debugging and error information, they also increase I/O latency. Therefore, QLogic does not recommend that you enable extended logging during normal operations.

Troubleshooting with Driver Logs

All messages from the QLogic FC adapter driver are logged in `/var/log/vmkernel`. These messages provide additional information about the driver operation: both normal and driver warnings. Messages from the driver are usually preceded by `scsi(%d)`, where `%d` is the adapter instance number.

The following example shows a snippet from a `vmkernel` file:

```
Nov 12 16:11:49 localhost vmkernel: 0:00:00:04.754
cpu1:1036)<6>qla2x00_set_info starts at address = 0x8b0000

Nov 12 16:11:49 localhost vmkernel: 0:00:00:04.754
cpu1:1036)<6>qla2x00: Found  VID=1077 DID=2432 SSVID=1077
SSDID=138 pdev=0x2869668

Nov 12 16:11:49 localhost vmkernel: 0:00:00:04.754
cpu1:1036)qla2x00_detect(0): subsys_vend=0x1077 subsys_dev=0x138

Nov 12 16:11:49 localhost vmkernel: 0:00:00:04.754
cpu1:1036)<6>scsi(0): 64 Bit PCI Addressing Enabled.

Nov 12 16:11:49 localhost vmkernel: 0:00:00:04.756
cpu1:1036)<7>scsi(0): Allocated 4096 SRB(s).

Nov 12 16:11:49 localhost vmkernel: 0:00:00:04.756
cpu1:1036)<7>PCI: Setting latency timer of device 1f:00.0 to 64

Nov 12 16:11:49 localhost vmkernel: <6>bnx2: vmnic0 NIC Copper
Link is Up, 100 Mbps full duplex

Nov 12 16:11:49 localhost vmkernel: 0:00:00:06.410
cpu1:1036)scsi(0): Waiting for LIP to complete...

Nov 12 16:11:49 localhost vmkernel: 0:00:00:07.196
cpu1:1036)<6>scsi(0): LOOP UP detected.

Nov 12 16:11:49 localhost vmkernel: 0:00:00:07.502
cpu1:1036)<6>qla2x00_fw_ready ha_dev_f=0x0
```

Table A-6 lists the more common driver log messages and provides an explanation of each.

Table A-6. Common Driver Log Messages

"qla2x00_set_info starts at address = *XXXXXXXX*"
Driver is reporting the starting address where the driver was loaded in case an oops occurs in the driver.
"qla2x00: Found VID=*XXXX* DID=*yyyy* SSVID=*ZZZZ* SSDID=*VVVV*"
Driver is reporting which adapter it has found during initialization.
"scsi(%d): Allocated *XXXXX* SRB(s)"
Driver is reporting the number of simultaneous commands that can be executed by the adapter. The `max_srbs` option (see Table 3-1 on page 3-7) can change this number.

Table A-6. Common Driver Log Messages (Continued)

`"scsi(%d): 64 Bit PCI Addressing Enabled"`
Driver is reporting that it has configured the adapter for 64bit PCI bus transfers.
`"scsi(%d): Verifying loaded RISC code..."`
Driver is reporting that it has verified the RISC code and it is running.
`"scsi(%d): Verifying chip..." extended"`
Driver is reporting that it has verified the chip on the adapter.
`"scsi(%d): Waiting for LIP to complete..."`
Driver is reporting that it is waiting on the firmware to become ready.
`"scsi(%d): LIP occurred, ..."`
Driver received a LIP async event from the firmware.
`"scsi(%d) LOOP UP detected"`
Driver received a loop up async event from the firmware.
`"scsi(%d) LOOP DOWN detected"`
Driver received a loop down async event from the firmware.
`"scsi(%d): Link node is up"`
Driver received a point-to-point async event from the firmware.
`"scsi%d: Topology - (%s), Host Loop address 0x0"`
Indicates the firmware connection type. `%s` will be one of the following: `FL-PORT`, `N-PORT`, `F-PORT`, `NL-PORT`, and host adapter loop ID.
`"scsi%d : QLogic XXXXXX PCI to Fibre Channel Host Adapter: ... " Firmware version: 4.04.06, Driver version 7.08vm62"`
Driver is reporting information discovered during its initialization. This information includes the board ID, firmware version, and driver version.
`"qla%d Loop Down - aborting ISP"`
Indicates driver is attempting to restart the loop by resetting the adapter. Usually done by the driver when sync is not detected by the firmware for a long time (4+ minutes), and usually means that the adapter port is not connected to the switch or loop.
`"scsi(%d): %s asynchronous Reset."`
Driver received an async reset event from the firmware. `%s` indicates the function name.
`"qla2x00: ISP System Error - mbx1=%x, mbx2=%x, mbx3=%x"`

Table A-6. Common Driver Log Messages (Continued)

Driver received an async ISP system error event from the firmware. Additional information follows the message (that is, mailbox values from the firmware).

`"scsi(%d): Configuration change detected: value %d."`

Driver received a change in connection async event from the firmware. Additional information follows the message (that is, mailbox 1 value from the firmware).

`"scsi(%d): Port database changed"`

Driver received a port database async event from the firmware.

`"scsi(%d): RSCN,..."`

Driver received a registered state change notification (RSCN) async event from the firmware. Additional information follows the message (that is, mailbox values from the firmware).

`"%s: Can't find adapter for host number %d\n"`

Indicates that the read from `/proc/scsi/qla2X00` did not specify the correct adapter host number. `%s` indicates the function name.

`"scsi(%d): Cannot get topology - retrying"`

Firmware return status indicating it is busy.

`"%s(): **** SP->ref_count not zero\n"`

Indicates a coding error. `%s` is the function name.

`"qla_cmd_timeout: State indicates it is with ISP, But not in active array"`

Indicates a coding error. `%s` is the function.

`"cmd_timeout: LOST command state = 0x%x\n"`

Indicates the command is in an undefined state. `0x%x` indicates the state number.

`"qla2x00: Status Entry invalid handle"`

Driver detected an invalid entry in the ISP response queue from the firmware. `%x` indicates the queue index.

`"%s(): **** CMD derives a NULL TGT_Q\n"`

Indicates the command does not point to an OS target.

`"scsi(%ld:%d:%d:%d): DEVICE RESET ISSUED.\n"`

Indicates a device reset is being issued to (`host:bus:target:lun`).

`"scsi(%ld:%d:%d:%d): LOOP RESET ISSUED.\n"`

Indicates a loop reset is being issued to (`host:bus:target:lun`).

Table A-6. Common Driver Log Messages (Continued)

"%s(): ** CMD derives a NULL HA\n"**
Indicates the command does not point to an adapter structure.
"%s(): ** CMD derives a NULL search HA\n"**
Indicates the command does not point to the adapter structure.
"scsi(%ld:%d:%d:%d): now issue ADAPTER RESET.\n"
Indicates an adapter reset is being issued to (`host:bus:target:lun`).
"scsi(%d): Unknown status detected %x-%x"
Indicates the status returned from the firmware is not supported. `%x-%x` is the `completion-scsi` statuses.
"scsi(%ld:%d:%d:%d): Enabled tagged queuing, queue depth %d.\n"
Indicates the queue depth for the (`host:bus:target:lun`).
"PCI cache line size set incorrectly (%d bytes) by BIOS/FW,"
Indicates a correction in the cache size. `%d` is the cache size.
"scsi(%d): Cable is unplugged..."
Indicates the firmware state is in `LOSS OF SYNC`; therefore, the cable must be missing.
"qla2x00: Performing ISP error recovery - ha=%p."
Indicates the driver has started performing an adapter reset.
"qla2x00_abort_isp(%d): ** FAILED ****"**
Indicates the driver failed performing an adapter reset.
"%s(%ld): RISC paused, dumping HCCR (%x) and schedule an ISP abort (big-hammer)\n"
Indicates the driver has detected the RISC in the pause state.
"scsi(%ld): Mid-layer underflow detected (%x of %x bytes) wanted "%x bytes...returning DID_ERROR status!\n"
Indicates an underflow was detected.
"%s(): Ran out of paths - pid %d"
Indicates there are no more paths to try for the request. `%s` is the function name and `%d` is the mid-level processor identifier (PID).
"WARNING %s(%d):ERROR Get host loop ID"
Firmware failed to return the adapter loop ID.

Table A-6. Common Driver Log Messages (Continued)

"WARNING qla2x00: couldn't register with scsi layer\n"
Indicates the driver could not register with the SCSI layer, usually because it could not allocate the memory required for the adapter.
"WARNING scsi(%d): [ERROR] Failed to allocate memory for adapter\n"
Indicates the driver could not allocate all the kernel memory it needed.
"WARNING qla2x00: Failed to initialize adapter\n"
Indicates that a previously occurring error is preventing the adapter instance from initializing normally.
"WARNING scsi%d: Failed to register resources.\n"
Indicates the driver could not register with the kernel.
"WARNING qla2x00: Failed to reserve interrupt %d already in use\n"
Indicates the driver could not register for the interrupt IRQ because another driver is using it.
"WARNING qla2x00: ISP Request Transfer Error"
Driver received a Request Transfer Error async event from the firmware.
"WARNING qla2100: ISP Response Transfer Error"
Driver received a Response Transfer Error asynchronous event from the firmware.
"WARNING Error entry invalid handle"
Driver detected an invalid entry in the ISP response queue from the firmware. This error will cause an ISP reset to occur.
"WARNING scsi%d: MS entry - invalid handle"
Driver detected a management server command timeout.

Debugging Using Proc Information

The driver `/proc/scsi/qla2300/<adapter_number>` provides debugging information that can help you troubleshoot problems with the driver. The `/proc` entry provides firsthand information about the current state of the driver.

Table A-8 lists the `/proc` output fields and provides a description of each.

Table A-7. Proc Output

Output Field	Description
`Boot Code Version`	Provides the version of the boot code present on the adapter. A version of `0.00` indicates that the boot code is corrupted or not present, and must be updated to perform a *boot from SAN* (see "Configuring Boot from SAN" on page 3-11).
`Number of ISP aborts`	Provides the number of times the system chip has been reset. Typically, whenever the system NVRAM or Flash is updated, the chip is reset to use the updated information. Otherwise, the chip is reset only if an error occurs. This number thus provides a close approximation of whether the chip has been operating properly.
`Host adapter FC link state`	Provides the connection state of the specified adapter. Possible values include the following: `READY:` Ready to perform I/Os. `UPDATE:` The topology is being re-discovered. `DEAD:` No connection. `DOWN:` Intermittent state, link temporarily down.
`NPIV Supported`	Indicates whether the given adapter node supports NPIV. If `yes`, the sub-section provides the maximum number of virtual ports supported.
`SCSI Device Information`	Provides the WWN of the given adapter.
`FC Port Information`	Lists all FC ports to which the specified adapter is connected. The last entry in this field indicates the login status of the port and whether the port is operating in *target* or *initiator* mode.
`SCSI LUN Information`	Lists all the LUNs visible to the specified adapter.

Collecting System Logs for Troubleshooting

To collect all relevant data on the running system into one `tar` file, execute the following command:

```
# vm-support
```

Frequently Seen Issues

Table A-8 lists some common issues and questions and provides tips for their resolution.

Table A-8. Frequently Seen Issues

Issue	Troubleshooting Tip
What is the SANsurfer FC HBA Manager password?	The password is *config*.
One specific adapter in the system has performance issues, while all other adapters in the system are okay.	Ensure that the adapter is plugged into a slot that allows the adapter to use the full bandwidth available. Also check that the bus the adapter is on is not shared by other high bandwidth PCI cards.
Errors similar to the following occur when the system is booted up. In addition, the ALT+Q QLogic BIOS banner is not shown; or FlashUTIL! is hanging when trying to flash the adapters. ■ `00180101 10772422 PCID device resource allocation failure` ■ `00180101 10772422 PCID device resource allocation failure`	There are too many devices taking up extended BIOS data area (EBDA) memory resources. To free up enough resources to allow the adapters to work properly, you can disable unused onboard services such as Wake on LAN, PXE boot, and so on.
Different target numbers can be assigned to the various storage devices for the adapters on the SAN. What is the correct method to assign persistent bindings?	QLogic recommends that you assign the target numbers of the devices in the same order on multiple adapters. For example, if controller A of an array is assigned to target 0 on HBA0, you should assign controller B of an array to target 0 on HBA1, and then continue this same methodology for all targets and adapters. Note that the driver and OS must be reloaded for the changes to take effect.
Rescanning an adapter port for target/LUN discovery.	`# esxcfg-rescan vmhba<X>` You can find the `vmhba` value corresponding to an adapter port by examining the `/var/log/vmkernel` log, where the adapter is assigned a `vmhba<X>` vector.

Notes

Glossary

adapter

The board that interfaces between the host system and the target devices. Adapter is synonymous with *host bus adapter (HBA)*, *host adapter*, and a*dapter board*.

arbitrated loop

A circular (ring) topology (versus point-to-point) where two or more ports can be interconnected, but only two ports can communicate at a time. All communication passes through all ports connected to the loop.

BIOS

Basic input output system (typically in Flash PROM). The program (or utility) that serves as an interface between the hardware and the operating system and allows booting from the adapter at startup.

boot code

The program that initializes a system or an adapter. Boot code is the first program to run when a system or a devices within a system, such as an adapters. FCode, BIOS, and EFI (enhanced firmware interface) are all forms of boot code for specific hardware/operating system environments.

Boot code for QLogic FC adapters is required if the computer system is booting from a storage device (disk drive) attached to the adapter. The primary function of the boot code is communication with the external boot device before the operating system is up and running. There are also secondary functions that can be performed by the boot code, including managing the setup for the adapter and initializing and testing the adapter's ISP.

boot device

The device, usually a the hard disk, that contains the operating system the BIOS uses to boot from when the computer is started.

cache

A temporary high-speed storage (memory) area where recently used or frequently accessed data is stored for rapid access, thus increasing the efficiency of processor operations.

Cache contents duplicate recently-used values stored on disk or elsewhere. Cache is quick to access relative to the original data which either takes more time due to slow disc-access time, or is slow to compute. Once data is stored in the cache, future use can be made by accessing the cached copy rather than re-finding or recomputing the original data, so that the average access time is lower.

- **L1 cache**—Primary (smallest) cache on the same chip as the processor.

- **L2 cache**—Secondary (larger) cache. Either on the processor chip or external to the processor.

device

A target, typically a disk drive. Hardware such as a disk drive, tape drive, printer, or keyboard that is installed in or connected to a system. In FC, a *target* device.

driver

The software that interfaces between the file system and a physical data storage device or network media.

The level structure for Windows XP Professional/2000/Windows Server 2003 drivers is as follows:

- *Class Driver*. The highest driver level. There is a separate class for disk, Ethernet, and so forth. This level handles all generic aspects of operations for that class.

- *Port Driver*. The middle driver level, which handles aspects of the operation specific to the port type; for example, there is a port driver for SCSI.

In Red Hat/SUSE Linux, the driver layers include:

- *SCSI Upper Layer.* This is the device management layer. It handles device-dependent tasks for devices, such as disks and tapes.

- *SCSI Middle Layer.* This is the SCSI traffic handling layer. It directs requests between the kernel and the SCSI.

- *SCSI Lower Layer.* This is the SCSI adapter driver. It communications directly to the SCSI adapter.

The structure for Solaris SPARC drivers includes:

- *Nexus Drivers.* Nexus drivers provide bus mapping and translation services to subordinate nexus and leaf devices. These include drivers for PCI-to-PCI bridges, PCMCIA adapters, and SCSI adapters.

- *Leaf Drivers.* Leaf drivers provide the traditional character and block driver interfaces for reading and writing data to storage and communication devices. These include drivers for peripheral devices, including QLA200/QLA2*xxx* adapters, disks, tapes, network adapters, and frame buffers.

- *Miniport Driver*. The lowest driver level and device specific. This level is usually supplied by the manufacturer as a companion to a physical device.

- *Monolithic Driver*. This level combines the functions of different driver levels in the same driver to increase performance.

- A*djunct Driver*. This level works along side a driver at the same level to increase performance.

In NetWare, the required drivers include:

- *Host Adapter Module (HAM)*. HAM is the driver component associated with the host adapter hardware. It provides the functionality to route requests to the bus where a specified device is attached.

- *Custom Device Module (CDM)*. The drive component associated with storage devices. It provides the functionality to build device-specific commands from I/O messages received from NetWare's Media Manager.

E_Port

Expansion port. A port in an FC switch that connects to another FC switch or bridge device by an inter-switch link. E_Ports are used to link FC switches to form a multi-switch fabric.

F_Port

The "Fabric" port in a FC fabric switch provides a point-to-point link attachment to a single N_Port. F_Ports are intermediate ports in virtual point-to-point links between end ports, for example N_Port to F_Port to F_Port to N_Port using a single FC fabric switch.

fabric

A fabric consists of cross-connected FC devices and switches.

fabric switch

Also, switched fabric. A fabric switch connects multiple devices from independent FC-arbitrated loops (FC-ALs) and point-to-point topologies into a fabric using FC switches.

***Fast!*UTIL**

QLogic *Fast!*UTIL FC adapter BIOS utility.

FCode

A type of boot code for Sun's SPARC platforms. See boot code and Flash.

FC

Fibre Channel is a high-speed serial interface technology that supports other higher layer protocols such as SCSI and IP.

FDMI

Fabric device management interface is a standard enabling the management of devices such as adapters through the fabric.

FL_Port

Fabric loop port. In an FC, the fabric switch is capable of FC Arbitrated Loop operations and is connected to one or more NL_Ports by an FC Arbitrated Loop. An FL_Port becomes a shared entry point for public NL_Port devices to an FC fabric. FL_Ports are intermediate ports in virtual point-to-point links between end ports that do not reside on the same loop, for example NL_Port to FL_Port to F_Port to N_Port through a single FC fabric switch.

Flash

Nonvolatile memory holding the boot code. At times, Flash and boot code are used interchangeably. (See boot code)

G_Port

Generic port. A port that can operate as either an E_Port or an F_Port. A G_Port can determine operating mode at switch port initialization, F_Port when an N_Port attachment is determined, E_Port when an E_Port attachment is determined. See E_Port, F_Port, FL_Port, L_Port, N_Port (Node Port), NL_Port (Node Loop Port)

HBA

Host bus adapter. See adapter.

HBA port

A port on the adapter board.

HBA port beacon

An LED on the adapter. Flashing it enables you to locate the adapter.

ioctl

Input/output control. A system call in Unix/Linux systems. Allows an application to control or communicate with a device driver outside usual read/write operations.

L_Port

Loop port. Does arbitrated loop functions and protocols. NL_Ports and FL_Ports are examples of loop-capable ports. See E_Port, F_Port, FL_Port, G_Port, N_Port (Node Port), NL_Port (Node Loop Port).

LIP

Loop initialization process. The initialization process in an arbitrated loop that occurs when the loop is powered up or a new device is added. One function of a LIP is to assign addresses. All data transmission on the loop is suspended during a LIP.

loopback

Diagnostic tool that routes transmit data through a loopback connector back to the same adapter.

LUN

Logical unit number, a subdivision of a SCSI target. It is the small integer handle that differentiates an individual disk drive or partition (volume) within a common SCSI target device such as a disk array.

Technically, a LUN can be a single physical disk drive, multiple physical disk drives, or a portion (volume) of a single physical disk drive. However, LUNs are typically not entire disk drives but rather virtual partitions (volumes) of a RAID set.

Using LUNs, the FC host can address multiple peripheral devices that may share a common controller.

media

Physical-layer information carriers. FC supports several different physical media: copper, multimode optical, and single-mode optical. All FC protocols are supported on all media.

N_Port (Node Port)

A port that connects by a point-to-point link to either a single N_Port or a single F_Port. N_Ports handle creation, detection, and flow of message units to and from the connected systems. N_Ports are end ports in virtual point-to-point links through a fabric, for example N_Port to F_Port to F_Port to N_Port using a single FC fabric switch. See also FL_Port.

N_Port ID virtualization

The ability for a single physical FC end point (N_Port (Node Port)) to support multiple, uniquely addressable, logical end points. With NPIV, a host FC adapter is shared in such a way that each virtual adapter is assigned to a virtual server and is separately identifiable within the fabric. Connectivity and access privileges within the fabric are controlled by identification of each virtual adapter and, hence, the virtual server using each virtual adapter.

NIC

Network interface controller.

NL_Port (Node Loop Port)

A port capable of arbitrated loop functions and protocols. An NL_Port connects through an arbitrated loop to other NL_Port and at most a single FL_Port. NL_Ports handle creation, detection, and flow of message units to and from the connected systems. NL_Ports are end ports in virtual point-to-point links through a fabric, for example NL_Port to F_Port to F_Port to N_Port using a single FC fabric switch. In the absence of a fabric switch FL_Port, NL_Ports can communicate with other NL_Ports in virtual point-to-point links through a FC_AL open loop circuit often through FC_AL (Arbitrated Loop) hub or loop switch devices. See: E_Port, F_Port, FL_Port, G_Port, N_Port (Node Port).

NPIV

See N_Port ID virtualization.

NVRAM

Non-volatile random access memory. A type of memory that retains data even when power is removed. Holds configuration settings. You can configure NVRAM settings manually or restore them from a file.

oops

A deviation from correct behavior of the Linux kernel that produces a certain error log. The kernel panic condition results from many kinds of oops, but others may allow continued operation with compromised reliability. When the kernel detects a problem, it prints an oops message and kills any offending process.

path

A path to a device is a combination of a adapter port instance and a target port as distinct from internal paths in the fabric network. A fabric network appears to the operating system as an opaque network between the adapter (initiator) and the target.

Because a path is a combination of an adapter and a target port, it is distinct from another path if it is accessed through a different adapter and/or it is accessing a different target port. Consequently, when switching from one path to another, the driver might be selecting a different adapter (initiator), a different target port, or both.

This is important to the driver when selecting the proper method of failover notification. It can make a difference to the target device, which might have to take different actions when receiving retries of the request from another initiator or on a different port.

PID (process identifier)

A number used by some operating system kernels (such as that of UNIX, Mac OS$^®$ X or Windows NT) to uniquely identify a process.

PLOGI (port login)

A port login occurs in a Fibre Channel SAN when two node ports establish a connection between each other (typically a device such as an FC adapter connecting to a switch).

point-to-point

Also FC-P2P. Two FC nodes directly connected (not in a loop).

port

Access points in a device where a link attaches. There are four types of ports, as follows:

- N_Port (Node Port)—an FC port that supports point-to-point topology.
- NL_Port (Node Loop Port)—an FC port that supports loop topology.
- FL_Port—a port in a fabric where an N_Port can attach.
- FL_Port—a port in a fabric where an NL_Port can attach.

port instance

The number of the port in the system. Each adapter may have one or multiple ports, identified with regard to the adapter as port 0, port 1 and so forth. to avoid confusion when dealing with a system containing numerous ports, each port is assigned a port instance number when the system boots up. So Port 0 on an adapter might have a port instance number of, for example, 8 if it is the eighth port discovered by the system.

reduced instruction set computer (RISC)

A computer microprocessor that performs fewer types of computer instructions, thereby operating at higher speeds.

RSCN

Registered state change notification (RSCN) is an FC fabric notification sent to all specified nodes when any major fabric changes occur. This notification allows nodes to immediately gain knowledge about the fabric and react accordingly.

SAN

Storage area network. Multiple storage units (disk drives) and servers connected by networking topology.

SCSI

Small computer system interface. A high-speed interface used to connect devices, such as hard drives, CD drives, printers, and scanners, to a computer. The SCSI can connect many devices using a single controller. Each device is accessed by an individual identification number on the SCSI controller bus.

target

The storage-device endpoint of a SCSI session. Initiators request data from targets. Targets are typically disk-drives, tape-drives, or other media devices. Typically a SCSI peripheral device is the target but an adapter may, in some cases, be a target. A target can contain many LUNs.

A target is a device that responds to a requested by an initiator (the host system). Peripherals are targets, but for some commands (for example, a SCSI COPY command), the peripheral may act as an initiator

Index